PORTAL

US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+abstract:exception +abstract:speculative

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **exception speculative**

Found **9** of **138,663**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 9 of 9

Relevance scale

1 Sentinel scheduling: a model for compiler-controlled speculative execution

Scott A. Mahlke, William Y. Chen, Roger A. Bringmann, Richard E. Hank, Wen-Mei W. Hwu, B. Ramakrishna Rau, Michael S. Schlansker

November 1993 **ACM Transactions on Computer Systems (TOCS)**, Volume 11 Issue 4

Full text available: pdf(2.26 MB) Additional Information: full citation, abstract, references, citings, index terms

Speculative execution is an important source of parallelism for VLIW and superscalar processors. A serious challenge with compiler-controlled speculative execution is to efficiently handle exceptions for speculative instructions. In this article, a set of architectural features and compile-time scheduling support collectively referred to as sentinel scheduling is introduced. Sentinel scheduling provides an effective framework for both compiler-controlled speculative executi ...

Keywords: VlIW processor, exception detection, exception recovery, instruction scheduling, instruction-level parallelism, speculative execution, superscalar processor

2 Unconstrained speculative execution with predicated state buffering

Hideki Ando, Chikako Nakanishi, Tetsuya Hara, Masao Nakaya

May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture**, Volume 23 Issue 2

Full text available: pdf(1.50 MB) Additional Information: full citation, abstract, references, citings, index terms

Speculative execution is execution of instructions before it is known whether these instructions should be executed. Compiler-based speculative execution has the potential to achieve both a high instruction per cycle rate and high clock rate. Pure compiler-based approaches, however, have greatly limited instruction scheduling due to a limited ability to handle side effects of speculative execution. Significant performance improvement is, thus, difficult in non-numerical applications. This paper ...

3 An out-of-order execution technique for runtime binary translators

Bich C. Le

October 1998 **Proceedings of the eighth international conference on Architectural support for programming languages and operating systems**, Volume 32 , 33 Issue 5 , 11

Full text available: pdf(1.04 MB) Additional Information: full citation, abstract, references, citings, index terms

A dynamic translator emulates an instruction set architccturc by translating source instructions to native code during execution. On statically-scheduled hardware, higher performance can potentially be achieved by reordering the translated instructions; however, this is a challenging transformation if the source architecture supports precise exception semantics, and the user-level program is allowed to register exception handlers. This paper presents a software technique which allows a translato ...

4 Sentinel scheduling for VLIW and superscalar processors
Scott A. Mahlke, William Y. Chen, Wen-mei W. Hwu, B. Ramakrishna Rau, Michael S. Schlansker
September 1992 **ACM SIGPLAN Notices , Proceedings of the fifth international conference on Architectural support for programming languages and operating systems**, Volume 27 Issue 9
Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings, index terms

Speculative execution is an important source of parallelism for VLIW and superscalar processors. A serious challenge with compiler-controlled speculative execution is to accurately detect and report all program execution errors at the time of occurrence. In this paper, a set of architectural features and compile-time scheduling support referred to as sentinel scheduling is introduced. Sentinel scheduling provides an effective framework for compiler-controlled speculative ex ...

5 Dynamic translation: The Transmeta Code Morphing™ Software: using speculation, recovery, and adaptive retranslation to address real-life challenges
James C. Dehnert, Brian K. Grant, John P. Banning, Richard Johnson, Thomas Kistler, Alexander Klaiber, Jim Mattson
March 2003 **Proceedings of the international symposium on Code generation and optimization: feedback-directed and runtime optimization**
Full text available: pdf(988.25 KB) Publisher Site Additional Information: full citation, abstract, references

Transmeta's Crusoe microprocessor is a full, system-level implementation of the x86 architecture, comprising a native VLIW microprocessor with a software layer, the **Code Morphing Software** (CMS), that combines an interpreter, dynamic binary translator, optimizer, and runtime system. In its general structure, CMS resembles other binary translation systems described in the literature, but it is unique in several respects. The wide range of PC workloads that CMS must handle gracefully in real ...

Keywords: binary translation, dynamic optimization, dynamic translation, emulation, self-modifying code, speculation

6 Integrated predicated and speculative execution in the IMPACT EPIC architecture
David I. August, Daniel A. Connors, Scott A. Mahlke, John W. Sias, Kevin M. Crozier, Ben-Chung Cheng, Patrick R. Eaton, Qudus B. Olaniran, Wen-mei W. Hwu
April 1998 **ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture**, Volume 26 Issue 3
Full text available: pdf(1.60 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Explicitly Parallel Instruction Computing (EPIC) architectures require the compiler to express program instruction level parallelism directly to the hardware. EPIC techniques which enable the compiler to represent control speculation, data dependence speculation, and predication have individually been shown to be very effective. However, these techniques have not been studied in combination with each other. This paper presents the IMPACT EPIC Architecture to address the issues involved in design ...

7 Speculative software management of datapath-width for energy optimization
Gilles Pokam, Olivier Rochecouste, André Seznec, François Bodin

June 2004 **ACM SIGPLAN Notices , Proceedings of the 2004 ACM SIGPLAN/SIGBED conference on Languages, compilers, and tools**, Volume 39 Issue 7
Full text available: pdf(609.97 KB) Additional Information: full citation, abstract, references, index terms

This paper evaluates managing the processor's datapath-width at the compiler level by means of exploiting dynamic narrow-width operands. We capitalize on the large occurrence of these operands in multimedia programs to build static narrow-width regions that may be directly exposed to the compiler. We propose to augment the ISA with instructions directly exposing the datapath and the register widths to the compiler. Simple exception management allows this exposition to be only speculative. In thi ...

Keywords: clock-gating, compiler, energy management, narrow-width regions, reconfigurable computing, speculative execution

8 Superscalar design: Cherry: checkpointed early resource recycling in out-of-order microprocessors
José F. Martínez, Jose Renau, Michael C. Huang, Milos Prvulovic, Josep Torrellas
November 2002 **Proceedings of the 35th annual ACM/IEEE international symposium on Microarchitecture**
Full text available: pdf(1.40 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms, review

This paper presents *CHeckpointed Early Resource RecYcling (Cherry)*, a hybrid mode of execution based on ROB and checkpointing that decouples resource recycling and instruction retirement. Resources are recycled early, resulting in a more efficient utilization. Cherry relies on state checkpointing and rollback to service exceptions for instructions whose resources have been recycled. Cherry leverages the ROB to (1) not require in-order execution as a fallback mechanism, (2) allow memory re ...

9 Efficient dynamic scheduling through tag elimination
Dan Ernst, Todd Austin
May 2002 **ACM SIGARCH Computer Architecture News**, Volume 30 Issue 2
Full text available: pdf(1.13 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

An increasingly large portion of scheduler latency is derived from the monolithic content addressable memory (CAM) arrays accessed during instruction wakeup. The performance of the scheduler can be improved by decreasing the number of tag comparisons necessary to schedule instructions. Using detailed simulation-based analyses, we find that most instructions enter the window with at least one of their input operands already available. By putting these instructions into specialized windows with fe ...

Keywords: dynamic scheduling, complexity-effective architecture, low-power architecture, last-tag prediction

Results 1 - 9 of 9

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore® RELEASE 1.7

Welcome
United States Patent and Trademark Office

» Se

Help FAQ Terms IEEE Peer Review

Quick Links

Welcome to IEEE Xplore

○ Home
○ What Can I Access?
○ Log-out

Tables of Contents

○ Journals & Magazines
○ Conference Proceedings
○ Standards

Search

○ By Author
○ Basic
○ Advanced

Member Services

○ Join IEEE
○ Establish IEEE Web Account
○ Access the IEEE Member Digital Library

Print Format

Your search matched **55** of **1046194** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

speculative <and> (fault <or> exception <or> trap <or> Search

□ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

1 **Error detection and handling in a superscalar, speculative out-of-or execution processor system**
Saxena, N.; Chien Chen; Swami, R.; Osone, H.; Thusoo, S.; Lyon, D.; Chang Dharmaraj, A.; Patkar, N.; Lu, Y.; Chia, B.;
Fault-Tolerant Computing, 1995. FTCS-25. Digest of Papers., Twenty-Fifth International Symposium on , 27-30 June 1995
Pages:464 - 471

[Abstract] [PDF Full-Text (676 KB)] **IEEE CNF**

2 **Using speculative execution for fault tolerance in a real-time systen**
Younis, M.F.; Tsai, G.; Marlowe, T.J.; Stoyenko, A.D.;
Engineering of Complex Computer Systems, 1995. Held jointly with 5th CSES 3rd IEEE RTAW and 20th IFAC/IFIP WRTP, Proceedings., First IEEE Internatio Conference on , 6-10 Nov. 1995
Pages:349 - 356

[Abstract] [PDF Full-Text (804 KB)] **IEEE CNF**

3 **Speculative execution exception recovery using write-back suppres**
Bringmann, R.A.; Mahlke, S.A.; Hank, R.E.; Gyllenhaal, J.C.; Hwu, W.W.;
Microarchitecture, 1993. Proceedings of the 26th Annual International Sympo on , 1-3 Dec. 1993
Pages:214 - 223

[Abstract] [PDF Full-Text (804 KB)] **IEEE CNF**

4 **Unconstrained speculative execution with predicated state bufferin**
Ando, H.; Nakanishi, C.; Hara, T.; Nakaya, M.;
Computer Architecture, 1995. Proceedings. 22nd Annual International Sympo

on , 22-24 June 1995
Pages:126 - 137

[Abstract] [PDF Full-Text (1232 KB)] **IEEE CNF**

---

5 **Code reordering and speculation support for dynamic optimization systems**
Nystrom, E.M.; Barnes, R.D.; Merten, M.C.; Hwu, W.W.;
Parallel Architectures and Compilation Techniques, 2001. Proceedings. 2001 International Conference on , 8-12 Sept. 2001
Pages:163 - 174

[Abstract] [PDF Full-Text (1120 KB)] **IEEE CNF**

---

6 **Cost-effective graceful degradation in speculative processor subsys the branch prediction case**
Almukhaizim, S.; Verdel, T.; Makris, Y.;
Computer Design, 2003. Proceedings. 21st International Conference on , 13-: Oct. 2003
Pages:194 - 197

[Abstract] [PDF Full-Text (248 KB)] **IEEE CNF**

---

7 **Register renaming and dynamic speculation: an alternative approac**
Moudgill, M.; Pingali, K.; Vassiliadis, S.;
Microarchitecture, 1993. Proceedings of the 26th Annual International Sympo on , 1-3 Dec. 1993
Pages:202 - 213

[Abstract] [PDF Full-Text (1032 KB)] **IEEE CNF**

---

8 **The Mips R10000 superscalar microprocessor**
Yeager, K.C.;
Micro, IEEE , Volume: 16 , Issue: 2 , April 1996
Pages:28 - 41

[Abstract] [PDF Full-Text (1184 KB)] **IEEE JNL**

---

9 **Three architectural models for compiler-controlled speculative exec**
Chang, P.P.; Warter, N.F.; Mahlke, S.A.; Chen, W.Y.; Hwu, W.W.;
Computers, IEEE Transactions on , Volume: 44 , Issue: 4 , April 1995
Pages:481 - 494

[Abstract] [PDF Full-Text (1104 KB)] **IEEE JNL**

---

10 **Eliminating exception constraints of Java programs for IA-64**
Ishizaki, K.; Inagaki, T.; Komatsu, H.; Nakatani, T.;
Parallel Architectures and Compilation Techniques, 2002. Proceedings. 2002 International Conference on , 22-25 Sept. 2002
Pages:259 - 268

[Abstract] [PDF Full-Text (426 KB)] **IEEE CNF**

---

11 **Spec-all: aggressive read/write access speculation method for DSI systems**
Furukawa, F.; Ootsu, K.; Yokota, T.; Baba, T.;
Innovative Architecture for Future Generation High-Performance Processors a Systems, 2003 , 17 July 2003
Pages:117 - 123

[Abstract] [PDF Full-Text (4275 KB)] **IEEE CNF**

---

12 **The Transmeta Code Morphing/spl trade/ Software: using speculat recovery, and adaptive retranslation to address real-life challenges**
Dehnert, J.C.; Grant, B.K.; Banning, J.P.; Johnson, R.; Kistler, T.; Klaiber, A., Mattson, J.;
Code Generation and Optimization, 2003. CGO 2003. International Symposiu on , 23-26 March 2003
Pages:15 - 24

[Abstract] [PDF Full-Text (391 KB)] **IEEE CNF**

---

13 **Multithreaded architectural support for speculative trace schedulin VLIW processors**
Agarwal, M.; Nandy, S.K.; v Eijndhoven, J.; Balakrishanan, S.;
Integrated Circuits and Systems Design, 2002. Proceedings. 15th Symposiur on , 9-14 Sept. 2002
Pages:43 - 48

[Abstract] [PDF Full-Text (251 KB)] **IEEE CNF**

---

14 **Evaluating low-cost fault-tolerance mechanism for microprocessor multimedia applications**
Sato, T.; Arita, I.;
Dependable Computing, 2001. Proceedings. 2001 Pacific Rim International Symposium on , 17-19 Dec. 2001
Pages:225 - 232

[Abstract] [PDF Full-Text (785 KB)] **IEEE CNF**

---

15 **Dependability analysis of a cache-based RAID system via fast distributed simulation**
Yiqing Huang; Kalbarczyk, Z.T.; Iyer, R.K.;
Reliable Distributed Systems, 1998. Proceedings. Seventeenth IEEE Symposi on , 20-23 Oct. 1998
Pages:254 - 260

[Abstract] [PDF Full-Text (192 KB)] **IEEE CNF**

---

1 2 3 4 **Next**

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search | Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting | No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top